WHAT IS CLAIMED IS:

1. A method for transforming application data structures into an XML document, the method comprising:

(a) writing an application program having data structures;

(b) writing a transformation program;

(c) executing the application program;

(d) executing the transformation program when called for by the application program to transform the data structures from the application program into an XML document; and

(e) sending the XML document to a recipient.

2. The method of claim 1, wherein executing the transformation program further comprises executing the transformation program on a dedicated virtual machine.

3. The method of claim 2, wherein the virtual machine runs on a web application server.

4. The method of claim 1, wherein writing the transformation program further comprises compiling the transformation program into a byte-code language.

5. The method of claim 4, wherein executing the transformation program further comprises executing the compiled transformation program on a dedicated virtual machine.

6. The method of claim 1, wherein the transformation program is written in markup language syntax.

7. The method of claim 1, wherein the transformation program includes a construct for reading a value within the data structures and writing the value to the XML document.

8. The method of claim 1, wherein the transformation program allows for literal XML elements, attributes, and text that appear within the transformation program to be written to the XML document.

9. The method of claim 1, wherein the transformation program includes a construct for specifying attributes to be written to the XML document.

10. The method of claim 1, wherein the transformation program includes a construct for declaring namespaces in the XML document.

11. The method of claim 1, wherein the transformation program includes a construct for skipping program instructions.

12. The method of claim 1, wherein the transformation program includes a construct for copying elements from data structures to the XML document.

13. The method of claim 1, wherein the transformation program includes a construct for calling another transformation program.

14. The method of claim 1, wherein the transformation program includes a construct for applying a transformation template associated with another transformation program called by the application program.

15. The method of claim 1, wherein the transformation program includes a construct for looping over data structures while creating the XML document.

16. The method of claim 1, wherein the transformation program includes a construct for executing conditional logic to create certain XML content within the XML document.

17. The method of claim 1, wherein the transformation program used to convert the data structures into the XML document can be used to convert the XML document back into data structures.

18. A method for transforming an XML document into application data structures, the method comprising:

(a) writing an application program configured to use data structures;

(b) writing a transformation program;

(c) executing the application program;

(d) executing the transformation program when called for by the application program to transform an XML document into data structures for the application program; and

(e) using the data structures within the application program.

19. The method of claim 18, wherein executing the transformation program further comprises executing the transformation program on a dedicated virtual machine.

20. The method of claim 19, wherein the virtual machine runs on a web application server.

21. The method of claim 18, wherein writing the transformation program further comprises compiling the transformation program into a byte-code language.

22. The method of claim 21, wherein executing the transformation program further comprises running the compiled transformation program on a dedicated virtual machine.

23. The method of claim 18, wherein the transformation program is written in a markup language syntax.

24. The method of claim 18, wherein the transformation program includes a construct for reading a value within the XML document and writing the value to the data structures.

25. The method of claim 18, wherein the transformation program includes literal XML elements that are matched in the XML document.

26. The method of claim 18, wherein the transformation program includes a construct

for matching the name of an attribute in the XML document.

27. The method of claim 18, wherein the transformation program includes a construct for matching a namespace declaration in the XML document.

28. The method of claim 18, wherein the transformation program includes a construct for skipping program instructions.

29. The method of claim 18, wherein the transformation program includes a construct for copying elements to the data structures from the XML document.

30. The method of claim 18, wherein the transformation program includes a construct for calling another transformation program.

31. The method of claim 18, wherein the transformation program includes a construct for applying a transformation template associated with another transformation program called by the application program.

32. The method of claim 18, wherein the transformation program includes a construct for looping over content from the XML document while creating the data structures.

33. The method of claim 18, wherein the transformation program includes a construct for executing conditional logic to create certain data structures.

34. The method of claim 18, wherein the transformation program used to convert the XML document into data structures can be used to convert the data structures back into the XML document.

35. An application system comprising:

a first process configured to execute an application program, wherein the application program is operable to use a set of data structures;

a second process configured to interpret a markup language document;

a transformation template configured to specify a symmetric mapping between the markup language document and the set of data structures; and

a transformation virtual machine running in association with the first process and operable to execute the transformation template;

wherein the transformation virtual machine is operable to perform a symmetric transformation between the markup language document and the set of data structures to allow the first process and the second process to exchange information.

36. The application system of claim 35 wherein the first process is an ABAP virtual machine running on an application server.

37. The application system of claim 35 wherein the second process is one of a client processor and a server processor configured to communicate with the application server.

38. The application system of claim 36 further comprising a database configured to communicate with the application server, wherein the database is operable to store at least one of XML data and non-XML data.

39. The application system of claim 36 wherein the application server is a web application server.